CHANGE REQUEST FORM

Version	State	Author	Date	Description
1	Draft	Marlowe Surop Jovelyn Lim	26 Dec 2019	Initial Document
2	Final	Jovelyn Lim	06 Oct 2020	Amend the error messages for invalid inputs based on the external library mapping.

<table>
<tr><th>Title</th><th colspan="3">Strike Price out of range rounding errors</th></tr>
<tr><td>Background</td><td colspan="2">At present, the Strike Price component is following the JavaScript IEEE 75 standard logic which means the acceptable maximum digit for the whole number is up to 20 total digits and up to 17 for decimal places. Any number input after the max digit will be rounded off. Also, any number less than 0 is not acceptable which means negative numbers are being rejected in creating the ISINs. Input value with alphanumeric or special characters are being deleted instead of getting rejected.

Certain amendment is required to support the Strike Price within Equity and Others asset classes. Such change is in response to RTS23 document with reference # ESMA 65-11-1193 dated November 04, 2019 of which details can be found here. At present, the behaviour of Strike Price is unpredictable and ISINs generated are not the same as per input value. Nonetheless, its behaviour must be dependent on Strike Price type.

RTS23 Annex – Table 3 states that required format for each type should be as per below:

Price Monetary Value – Total digit of 18 and up to 13 decimal places
Price Percentage – Total digit of 11 and up to 10 decimal places
Price Yield – Total digit of 11 and up to 10 decimal places
Price Basis Points – Total digit of 11 and up to 10 decimal places
No Price – “PNDG”

Input values should accept both positive and negative numbers. Decimal places should be separated by “.”, and negative numbers with prefix “- “. Also, values should be rounded-off and not truncated.</td><td><table><tr><td>JIRA</td><td>DSB-088</td></tr><tr><td>Type</td><td>Validation</td></tr><tr><td>Owner</td><td>Marlowe Surop
Jovelyn Lim</td></tr><tr><td>Version</td><td>2</td></tr><tr><td>State</td><td>Final</td></tr></table></td></tr>
<tr><td>Terms of Reference</td><td>Scope</td><td colspan="2">The impact of this change is for the asset classes with Strike Price attribute. In this case the asset classes that will have an effect will be Equity.Options, Other.Options and Other.Other.</td></tr>
<tr><td></td><td>Dependencies</td><td colspan="2">Current state of the strike price does not include Strike price type yet whether the numeric value is in monetary, Yields, Percentage or basis points and or PNDG if no price available.

As per DSB 48 – strike price type such as Monetary Value, Yields, Percentage, Basis Points or PNDG shall be added as a requirement in creating ISINS. Once the strike price type has been implemented, numerical parameters will follow.</td></tr>
<tr><td>Requirements</td><td colspan="3">The Strike price attribute will accept numeric input with regards to the strike price type chosen e.g. Monetary Value, Percentage, Yields or Basis Points. Maximum character will apply based on the Strike price type and RTS 23 documentation.

RTS 23 documentation format (2.3.4.2.31Strike Price):</td></tr>
</table>

	Price Monetary Value: (18/13) Price Percentage or Yields: (11/10) Price Basis Points: (18/17) No Price: 'PNDG' In addition, Strike price attribute should have below additional acceptance criteria depending on the attribute type: A. If the Strike price type is "Monetary value", below conditions should apply: • Input value will accept up to 18 total digits with possibility of decimal presentation of up to 13 digits. • Numerical field should accept both positive and negative values. • For fraction digits, decimal places should be separated by "." (full stop), and negative numbers with prefix "- ". • If the input values reached the maximum limit, values should be rounded off and not truncated. • If the value more than one decimal point, contains non-numerical value or has special characters, the input should be rejected and provide an error message "Error: Strike Price contains invalid input value" B. If the Strike price type is "yields", below conditions should apply: • Input value will accept up to 11 total digits with possibility of decimal presentation of up to 10 digits. • Numerical field should accept both positive and negative values. • For fraction digits, decimal places should be separated by "." (full stop), and negative numbers with prefix "- ". • If the input values reached the maximum limit, values should be rounded off and not truncated. • If the value more than one decimal point, contains non-numerical value or has special characters, the input should be rejected and provide an error message "Error: Strike Price contains invalid input value" C. If the Strike price type is "percentage", below conditions should apply: • Input value will accept up to 11 total digits with possibility of decimal presentation of up to 10 digits. • Numerical field should accept both positive and negative values. • For fraction digits, decimal places should be separated by "." (full stop), and negative numbers with prefix "- ". • If the input values reached the maximum limit, values should be rounded off and not truncated. • If the value more than one decimal point, contains non-numerical value or has special characters, the input should be rejected and provide an error message "Error: Strike Price contains invalid input value" D. If the Strike price type is "basis points", below conditions should apply: • Input value will accept up to 18 total digits with possibility of decimal presentation of up to 17 digits. • Numerical field should accept both positive and negative values. • For fraction digits, decimal places should be separated by "." (full stop), and negative numbers with prefix "- ". • If the input values reached the maximum limit, values should be rounded off and not truncated. • If the value more than one decimal point, contains non-numerical value or has special characters, the input should be rejected and provide an error message "Error: Strike Price contains invalid input value" E. If the Strike price is "no price", below conditions should apply: • Input value should be "PNDG". • If input value is not "PNDG", request should be rejected with an error message "Error message: Strike price is invalid for no price type"

<table>
<tr><td>Technical Considerations</td><td>Upon checking with the Developers there are no technical considerations as of the moment and requirements mentioned can be executed in the GUI.</td></tr>
<tr><td>Change Details</td><td>
For each of the in-scope templates, the strike price logic will apply based on the type provided on below:
a. For Valid Input Values:
• Strike price value should be limited in Strike Price type (Table below).
<table>
<tr><th>Strike Price Type</th><th>Maximum Digit</th><th>Maximum Decimal</th></tr>
<tr><td>Monetary Value</td><td>18</td><td>13</td></tr>
<tr><td>Yield</td><td>11</td><td>10</td></tr>
<tr><td>Percentage</td><td>11</td><td>10</td></tr>
<tr><td>Basis Points</td><td>18</td><td>17</td></tr>
<tr><td>No Price</td><td>N/A</td><td>N/A</td></tr>
</table>
• For decimal digits, values should be separated by “.” (full stop), and negative numbers with prefix “- “at the beginning.
• Strike Price value should be represented by numbers only.
• Input Value should accept both positive “+”and negative”- “values.
• Input value should accept “0” value
• If Strike Price type is “No Price”, then “PNDG” is the populated value in the Strike Price field.
b. For Invalid Input Values:
• Input Value should not contain alphanumeric, special characters or more than one decimal point.
• Any value for “No Price” type will be rejected other than “PNDG”
c. Behaviour for Valid Input Values:
• If input value is less than the total maximum digits or less than maximum digits after the decimal.
• If the input values reached the maximum limit, values should be rounded off from the last maximum digit and not truncated (see sample below).
123059385.9548304459582758-> 123059385.954830446
OR
1.2345678923568964783434355 -> 1.2345678923569
• “- “should be at the beginning of the input value
• Prefix “- “and “.” (full stop) are not counted in the maximum total digits
• If input value has prefix “+” at the beginning, value will be accepted but removing the “+” sign.
• Only “PNDG” value is accepted for No Price type
d. For Input values with Incorrect format
For Equity Standard Template
<table>
<tr><th>Request</th><th>Response in GUI</th><th>Response in Rest</th></tr>
<tr><td>If input value is more than maximum characters with no decimal points.
Sample:
632648923648274902374 0</td><td>Error: /Attributes/StrikePriceType/StrikePrice: numeric instance is greater than the required maximum (maximum: 1000000000000000000, found: 632648923648274902374 0)</td><td>"responseCode": 400, "message": "/Attributes/StrikePriceType/S Price: numeric instance is great than the required maximum (maximum: 1000000000000000000, found 632648923648274902374 0)",</td></tr>
</table>
</td></tr>
</table>

If input value is more than maximum characters in front of decimal point. Sample: 6326489236482749023740.666970909	Error: /Attributes/StrikePriceType/StrikePrice: numeric instance is greater than the required maximum (maximum: 1000000000000000000, found: 6326489236482749023740.666970909)	"responseCode": 400, "message": "/Attributes/StrikePriceType/StrikePrice: numeric instance is greater than the required maximum (maximum: 1000000000000000000, found: 6326489236482749023740.666970909)",
If input value is more than maximum characters but in front of the input value is less than the maximum digits, keep the integers and round off the decimal points up to the total maximum digits. 12846274901.21232731063186381 62	Monetary: 12846274901.2123273 Yields: 12846274901 Percentage: 12846274901 Basis Point: 12846274901.2123273	Monetary: 12846274901.2123273 Yields: 12846274901 Percentage: 12846274901 Basis Point: 12846274901.2123273

For Non-standard template

Request	Response in GUI	Response in Rest
If input value is more than maximum characters with no decimal points. Sample: 6326489236482749023740	Error: /Attributes/StrikePriceType: instance failed to match exactly one schema (matched 0 out of 5)	"responseCode": 400, "message": "/Attributes/StrikePriceType: instance failed to match exactly one schema (matched 0 out of 5
If input value is more than maximum characters in front of decimal point. Sample: 6326489236482749023740.666970909	Error: /Attributes/StrikePriceType: instance failed to match exactly one schema (matched 0 out of 5)	"responseCode": 400, "message": "/Attributes/StrikePriceType: instance failed to match exactly one schema (matched 0 out of 5
If input value is more than maximum characters but in front of the input value is less than the maximum digits, keep the integers and round off the decimal points up to the total maximum digits. 1.5315237863264892364823487208472307490237490237 40	Monetary: 1.5315237863265 Yields: 1.5315237863 Percentage: 1.5315237863 Basis Point: 1.53152378632648924	Monetary: 1.5315237863265 Yields: 1.5315237863 Percentage: 1.5315237863 Basis Point: 1.5315237863264

e. Behaviour for Invalid Input Values:

For Equity Standard Template

Request	Response in GUI	Response in Rest
If input value contains special characters -> 1.32312321!@23131	Value will be accepted but removing the special character in the input 1.3231232123131	"responseCode": 500, "message": "Something went wrong.",
If input value contains alphanumerical - > 1.2321423f32434	Value will be accepted but removing the letters in the input 1.232142332434	"responseCode": 500, "message": "Something went wrong.",

	If input value has more than one decimal point or more than one " -" sign Sample: 1.232.42354454 -1.243425-2434234 | Error: /Attributes/StrikePriceType/StrikePrice: instance type (null) does not match any allowed primitive type (allowed: ["integer","number","string"]) | "responseCode": 500, "message": "Something went wrong.",

For Non-standard template

Request	Response in GUI	Response in REST
If input value contains special characters -> 1.32312321!@23131	Value will be accepted but removing the special character in the input 1.3231232123131	"responseCode": 500, "message": "Something went wrong
If input value contains alphanumerical - > 1.2321423f32434	Value will be accepted but removing the letters in the input 1.232142332434	"responseCode": 500, "message": "Something went wrong
If input value has more than one decimal point or more than one " -" sign Sample: 1.232.42354454 -1.243425-2434234	Error: /Attributes/StrikePriceType: instance failed to match exactly one schema (matched 0 out of 5)	"responseCode": 500, "message": "Something went wrong
If input value in REST/API for "No Price" type is not "PNDG"	N/A	"responseCode": 400, "message": "/Attributes/StrikePriceType: instan failed to match exactly one schema (matched 0 out of 5)",

Change Impact

- Equity/ Option/ Basket
- Equity/ Option/ Non_Standard
- Equity/ Option/ Single_Index
- Equity/ Option/ Single_Name
- Other/ Option/ Non_Standard
- Other/ Other/ Non_Standard

Backward Compatibility

As per the extract starting Nov 2017 based on Equity Strike Price attribute; findings are as follows:

1.Strike price input with "e" value in both request and record template

Template Equity/Option/Single_Index: 8 records (0% of total): 8 new/updated, 0 expired.

Template Equity/Option/Single_Name: 1 record (0% of total): 1 new/updated, 0 expired.

2. Strike price with more than 18 total digits

Template Equity/Option/Non_Standard: 28 records (0.00% of total): 13 new/updated, 15 expired.

Template Equity/Option/Single_Index: 9 records (0.00% of total): 8new/updated, 1 expired.

Template Equity/Option/Single_Name: 10 records (0.00% of total): 0 new/updated, 10 expired.

3. Strike price input with "0" value in both request and record template

	Template Equity/Option/Basket: 7798 records (10% of total): 4611 new/updated, 3187 expired. Template Equity/Option/Non_Standard: 1785 records (1% of total): 1433 new/updated, 352 expired. Template Equity/Option/Single_Index: 14803 records (4% of total): 5516 new/updated, 9287 expired. Template Equity/Option/Single_Name: 72257 records (11% of total): 17298 new/updated, 54959 expired. 4.No negative output values found. 5. Strike price with exponential record template Template Equity/Option/Single_Index: 2 records (0% of total): 1 new/updated, 1 expired. Template Equity/Option/Single_Name: 16 records (0.001% of total): 0 new/updated, 16 expired. As per analysis, below should be taken into considerations when adding the Strike price type: • End users must be able to search previously created ISINS in both the GUI and API. • For the REST/API, there is "retrieve no create" functionality where users fill up all the required attributes and DSB will retrieve the ISIN with the same details but will not create a new one. Once new attribute has been introduced this will not be applicable to asset classes affected given that the Strike price type will be part of the required attribute hence will prevent user to retrieve the ISIN. Suggest workaround to retrieve the ISIN is via "search by attributes/ asset classes" to validate if such details exist in DSB or not. • In relation to DSB-048, if a user is not informed whether the ISIN is created or not and proceed in creating an ISIN with same attributes in the past but with a Strike price type, it will proceed user to create a new one.
Documentation	The following DSB documents are to be updated: • DSB UAT Product Definitions here ➢ Table 4 – Attribute Data Dictionary ➢ Table 8 – Appendix II - Validations • DSB PROD Product Definitions here ➢ Table 4 – Attribute Data Dictionary ➢ Table 8 – Appendix II – Validations • ANNA-DSB website, FAQ for Strike price here
References	https://www.esma.europa.eu/sites/default/files/library/esma65-11-1193_firds_reference_data_reporting_instructions_v2.1.pdf